ACKNOWLEDGEMENT

We would like to express our sincere gratitude and deep regards to our faculty Mrs. Manali Sarkar, Assistant professor, Computer Science & Engineering (Data Science) Department for her guidance, monitoring and constant encouragement throughout the course of the project. Your useful advice and suggestions were really helpful to us during the journey of this project. We are obliged to our project team members for the valuable cooperation, coordination and information provided by them in their respective fields. We are grateful for their cooperation during the period of our assignment.

Our sincere thanks to our project coordinator Mr. Subir Bhadra, Assistant Professor, Computer Science & Engineering (Data Science) Department, and Prof. Avijit Bose (HOD. Department of Computer Science & Engineering, MCKV Institute of Engineering) for their guidance and encouragement that helped us to bag an opportunity of working at a research work oriented project.

Abstract or summary of the project

The project focuses on developing an intelligent travel route suggestion system to assist visitors in navigating from their source to their destination. With increasing travel complexities, providing optimal route recommendations based on past experiences can significantly enhance travel efficiency and satisfaction. Travelers often face challenges such as unexpected difficulties, inefficient routes, and lack of personalized guidance. The project aims to address these issues by leveraging traveler feedback and patterns to suggest the best possible routes and foresee potential difficulties. Utilizing artificial intelligence, the system learns from historical travel data and user feedback. It uses a logistic regression model and neural networks to analyse textual feedback and quantify route difficulty based on various parameters like road conditions, weather, and traffic.

The system updates its recommendations in real-time, ensuring they are tailored to the current user's preferences and past experiences. This intelligent route suggestion system enhances travel experiences by providing accurate, personalized recommendations and difficulty predictions. By continuously learning from user data, it ensures safe, efficient, and enjoyable journeys, addressing common travel challenges and promoting informed travel decision.

Contents

1. Introduction:

- The main objective of the project is to help visitors travelling from their source to destination with best possible route. This system will consider route followed by most travelers and difficulties faced by travelers for various routes.
- The project learn from traveler experiences and use this information in guiding future travelers to tell which route will be preferable for them. This project can be developed for a country to guide the visitors for specific cities they want to visit.

2. Scope of the work:

- The main aim of using personalization techniques is to generate customized recommendation according to the user preferences and interests. The recommender system has an objective to filter unwanted information and to provide specific results for the particular user. In the travel recommender systems, proposed model learns the user preferences and generates places of attractions according to the user interests. This project focuses on the recommender systems and their application in tourism. To make this project useful to all, including new readers of recommender systems, it covers topics from evolution to applications along with the challenges in it.
- This project contributes clear review of recommender systems published in scientific journals and conferences with a special focus on travel recommender systems. These systems are analysed through the recommendation mechanism, interface, data source, and functionalities used.

3. Survey of the Previous Works:

- An-Jung Cheng et al. introduce a novel approach to travel route recommendation systems by incorporating people's attributes extracted from photos, such as gender, age, and race. Their research reveals that these attributes significantly influence decision-making regarding travel landmarks and paths. Leveraging demographic data from photos enhances the accuracy of landmark selection, path planning, and personalized travel recommendations, thus improving user experience and satisfaction.

- Takeshi Kurashima presents a travel route recommendation method that utilizes photographers' historical data from Flickr. The recommendation process is based on a photographer behaviour model, which estimates the probability of a photographer visiting a landmark. By analysed the photographers' past activities and preferences, the system recommends routes that align with their interests and behaviour, enhancing the relevance and effectiveness of travel suggestions.

d. Hardware and Software Details:

i. Only mention the additional hardware (if required):

1. **Name of the Hardware:** CPU, RAM, SSD, HDD, Motherboard, Monitor, Keyboard, Mouse etc.
2. **Purpose of the Hardware:**

- CPU: Performs calculations for the model training and prediction.
- RAM: Stores data and program instructions during execution.
- Graphics Processing Unit (GPU) (Optional): Can accelerate training if available and supported by libraries.
- SSD/HDD: Stores the dataset and trained model.
- Motherboard: Connects all the hardware components together and allows them to communicate.
- Monitor: Displays visual information.
- Keyboard and Mouse: Provide user input to the computer.

3. **Specification:**

- For CPU: Processor brand, number of cores, clock speed, cache size.
- For RAM: 16GB of RAM is recommended for smooth operation, especially if we're working with large datasets.
- For Storage (SSD/HDD): Capacity (GB, TB), type (SATA, NVme).
- For Monitors: Screen size, resolution, refresh rate, panel type (IPS, TN).
- GPU (optional): a mid-range Nvidia GeForce GTX or AMD Radeon RX series GPU can use.

4. **Model/ Built:** For this we use Logistic Regression Model and we'll build using TensorFlow. It will learn from our text data to classify next text entries.
5. **Price (Approx):**

- **CPU:** $150 - $500 (mid-range i5/Ryzen 5).
- **RAM:** $80 - $200 (16GB DDR4).
- **Storage:** $50 - $200 (256GB SSD).
- **GPU (Optional):** $200 - $500 (mid-range Nvidia GTX/AMD RX).

ii. Only mention those Software (Packages/ Tools) which will be used other than OS

1. **Name of the Package/ Tool:** Anaconda, TensorFlow, Python, Database (SQL, MySQL).
2. **Purpose of use:**
    - **Anaconda:**

    a. **Purpose:** Simplifies environment setup and provides essential data science libraries.

    - **TensorFlow:**

    b. **Purpose:** Develop and train machine learning models (logistic regression in this case).

    - **Python:**

    c. **Purpose:** Scripting, data manipulation, model development and interaction with TensorFlow.

Optional Database (Choose one):

- **MySQL (Free):**
    - **Purpose:** Store and manage visitor history data (free and open-source alternative).

3. Licensed/ Free / Open Sourced (with source link):

Free and Open Source (with source link):

- **Anaconda:** Free data science distribution (https://www.anaconda.com/)
    - Source Code: Not directly applicable as Anaconda is a pre-built package collection, but the source code for included packages (like Python, NumPy) is generally available.
- **TensorFlow:** Open-source library for numerical computation and machine learning (https://www.tensorflow.org/)
    - Source Code: https://github.com/tensorflow
- **Python:** Free and open-source general-purpose programming language (https://www.python.org/)
    - Source Code: https://github.com/python

Optional Database Software:

- **MySQL:** Free and open-source relational database management system (https://www.mysql.com/)
    - Source Code: https://github.com/mysql

iii. Dataset to be used (if any):

1. **Source of the dataset:** Kaggle
2. **Type of data:** Photo metadata from Flickr [1] using the site's public API.
3. **Number of total data and features:** The crawled data consists of 696,394 photographs and their associated metadata, which were taken by 71,718 unique users. All were taken in the East Coast and the West Coast of the United States between January 1st, 2006 and June 31st, 2009.

References:

- Nikhitha Pulmamidi1, Rajanikanth Aluvalu1 and V Uma Maheswari1, "Intelligent Travel Route Suggestion System Based on Pattern of Travel And Difficulties", IOP Conference Series:Materials Science and Engineering, vol. 1042, 13 pages, 2020.
- Logesh Ravi, Subramaniyaswamy Vairavasundaram, "A Collaborative Location Based Travel Recommendation System through Enhanced Rating Prediction for the Group of Users", Computational Intelligence and Neuroscience, vol. 2016, Article ID 1291358, 28 pages, 2016.
- Mohamed Barouche, Mehdi Boutaounte, “Personalized Travel Recommendation Systems: A Study of Machine Learning Approaches in Tourism”, Journal of Artificial Intelligence, Machine Learning and Neural Network, ISSN: 2799-1172, Vol: 03, No. 03, 2023.
- V. K. Muneer, K. P. Mohamed Basheer, “The evolution of travel recommender systems: A comprehensive review”, Malaya Journal of Matematik, Vol. 8, No. 4, 1777-1785, 2020.
- Junjie SUN, Chenyi ZHUANG, “User Transition Pattern Analysis for Travel Route Recommendation”, IEICE Transactions on Information and Systems, Vol. E102-D NO. 12, DECEMBER 2019.

Proposed works for the Seventh Semester:

The 7th semester will lay the groundwork for our travel route recommendation system. Here's a detailed breakdown of the key activities:

1. User Data Collection - Capturing Travel Preferences:

- **Designing User Surveys:**
    - Use Python libraries like Forms or SurveyMonkey to create user surveys or questionnaires.
    - Ensure the surveys are engaging and capture relevant information:
        - **Demographics:** Age, gender (optional) to understand user groups.
        - **Travel Preferences:**
            - Interest-based questions: Include multiple-choice options for travel interests (history, adventure, relaxation) or use Likert scales (strongly dislike to strongly like) to gauge preferences.
            - Budget: Allow users to select their preferred budget range for a trip.
            - Difficulty Level: Provide options like "easy," "moderate," or "challenging" to understand user preferences for physical accessibility and cultural complexities.
        - **Past Travel History:**
            - Use multiple-choice options or open-ended questions to capture locations visited previously.
            - Consider including options for users to rate their experiences at those locations (loved it, okay, wouldn't recommend).
            - Gather information on activities enjoyed at past destinations (e.g., hiking, museum visits).
- **Collecting Landmark Data - Building Your Route Database:**
    - **Website Scraping:** Utilize Python libraries like Beautiful Soup or Selenium to automate data collection from travel websites.
        - Focus on websites with comprehensive information on travel destinations and landmarks.
        - Consider partnering with travel agencies for access to their data (with proper agreements).
    - **Essential Landmark Details:**

- Location (city, country, GPS coordinates): Crucial for route planning and visualization.
- Type of Landmark: Categorize landmarks (museum, historical site, natural wonder).
- Difficulty Level: Similar to user preferences, categorize landmarks based on accessibility and cultural challenges.
- Popularity Ratings: Scrape average user ratings and estimated visitor numbers to gauge landmark popularity.

2. Data Preprocessing and Storage - Ensuring Clean Data:

- **Cleaning User Data**: Use Python libraries like Pandas to clean and organize user survey data.
  - Handle missing values (e.g., impute missing budget data with an average).
  - Standardize formats (e.g., convert all dates to a consistent format).
  - Remove irrelevant information (e.g., free-text comments outside specific question prompts).
- **Preparing Landmark Data**: Clean and organize scraped landmark data using Pandas.
  - Address inconsistencies (e.g., standardize landmark names).
  - Handle missing values (e.g., impute missing difficulty levels with an average).
  - Convert categorical data (landmark types) to numerical representations suitable for machine learning models.
- **Database Storage:** Use libraries like SQL Alchemy to store cleaned user and landmark data in your SQL Server database.
  - Create separate tables for users, landmarks, and potentially a table linking user travel history with specific landmarks (including ratings/activities).

3. Collaborative Filtering Model Development - Learning User Similarities:

- **Understanding CF Algorithms:**
  - **User-based CF**: This works by identifying users with similar travel preferences (based on survey data) and recommending routes popular among those similar users.

    - **Item-based CF**: This focuses on the similarities between landmarks themselves (considering type, difficulty, location). It recommends routes containing landmarks similar to those a user has enjoyed in the past.
- **Choosing the Algorithm:** The choice depends on your data structure. If user data is rich with travel preferences, user-based CF might be suitable. If landmark details are comprehensive, item-based CF could be a good option.
- **Implementation in Python:** Use machine learning libraries like scikit-learn or Surprise. These libraries provide pre-built implementations of CF algorithms. You'll need to feed your user-landmark interaction data (e.g., ratings, visit history) into the chosen algorithm for training.
- **Training the Model:** The CF algorithm will learn relationships between users or landmarks based on the training data. It will be able to predict the likelihood of a user choosing a particular route based on its similarity to routes preferred by similar users in the past.

4. Content-based Filtering Logic - Matching Preferences to Routes:

- **Rule-based Approach:** Develop a set of rules in Python that link user preferences (interests) to specific landmark types. For example:
    - Rule: If a user indicates an interest in history, recommend routes containing museums and historical sites.
    - Rule: If a user prefers a relaxing vacation, recommend routes with beaches and resorts.

5. Overall Working Principle:

1. User inputs preferences through the prototype.
2. The CF model retrieves routes popular among similar users based on past interactions.
3. The CBF logic suggests routes containing landmarks that align with user preferences.

Proposed Works for the Eighth Semester:

Building Upon the Foundation: 8th Semester Activities for our Travel Route Recommendation System:

In The 7th semester we laid the groundwork for our travel route recommendation system. Now, in the 8th semester, we can focus on enhancing functionalities, refining recommendations, and exploring innovative features. Here's a breakdown of key activities:

1. Refining Recommendation Models:

- **Evaluation:** Conduct user studies to assess the accuracy and effectiveness of your CF and CBF models.
    - Gather feedback on the quality and relevance of recommended routes.
    - Analyses user behaviour data to understand how users interact with recommendations.
- **Model Improvements:** Based on user feedback and analysis, explore ways to improve your models:
    - **CF:**
        - Consider incorporating matrix factorization techniques for more complex user-landmark relationships.
        - Explore hybrid CF approaches that combine user-based and item-based CF for potentially richer recommendations.
    - **CBF:**
        - Refine your rule-based approach to better capture nuanced user preferences.
        - If you implemented a machine learning model, explore using more advanced algorithms like Support Vector Machines or Neural Networks for improved prediction accuracy.

2. Real-time Data Integration:

- Enhance your system's ability to adapt to current conditions by integrating real-time data feeds:
    - **Weather data:** Recommend routes based on weather conditions at potential destinations (e.g., avoiding rainy regions for hiking trips).

- **Event schedules:** Suggest routes that incorporate upcoming events or festivals a user might be interested in.

3. User Feedback and Personalization:

- **Feedback Mechanism:** Implement a system for users to provide feedback on recommended routes. This can be through ratings, comments, or indicating preferred/disliked suggestions.
- **Personalized Learning:** Utilize user feedback to personalize recommendations over time.
    - The system can learn a user's evolving preferences and adjust future recommendations accordingly.

4. Gephi (GY) for Recommendation Visualization:

- **Export Data:** Extract user data, landmark details, and recommended routes from your database in a format compatible with Gephi (e.g., CSV).
- **Network Creation:** In Gephi, create nodes representing users and landmarks. Connect users to recommended landmarks with edges.
- **Visualization Enhancements:**
    - Use node and edge colors or sizes to represent user preferences, landmark types, or recommendation strength.
    - Analyses the network to identify clusters of users with similar preferences and popular routes within those clusters. This can help refine your recommendations further.

5. Additional Explorations (Optional):

- **Multi-modal Travel Planning:** Develop logic to suggest routes that incorporate different travel modes (walking, cycling, public transport) based on user preferences and landmark locations.
- **Sustainability Integration:** Explore ways to promote eco-friendly travel practices within your system. This could involve highlighting routes that minimize carbon footprint or recommending destinations with responsible tourism initiatives.

Outcomes of the 8th Semester:

- A refined travel route recommendation system with improved accuracy and user experience.
- A system that can adapt to real-time data and user feedback for personalized recommendations.